福井大学 工学部 研究報告 第49巻 第1号2001年3月
Mem. Fac. Eng. Fukui Univ., Vol. 49, No. 1 (March 2001)

遺伝的プログラミングをもちいた戦略知識の進化的獲得

弓削 孝文* 白井 治彦** 西野 順二** 小高 知宏** 小倉 久和**

Evolutional Acquisition of a Strategy Using Genetic Programming

Takafumi YUGE*, Haruhiko SHIRAI, Junji NISHINO**, Tomohiro ODAKA** and Hisakazu OGURA****

(Received February 28, 2001)

In this paper, we have been investigating an evolutional acquisition method of a strategy for the repeated janken game. We define a strategy for the repeated janken game as a numerical function. We tried to acquire function formed strategy of an opponent using Genetic Programming(GP). Gene evolved according to evaluation of win or lose rate. Thus, we consider to acquire elite gene that can win opponent in highly rate. The results, we could acquire an effective strategy against simple automatic players. In acquired gene there are models of opponent strategies. In addition, we investigated to be acquired human's strategy. As a result, we could get effective strategies against human.

Key Words : GA, GP, Learning, Evolution

1 はじめに

相手の考え方を知ることは, 交渉の場において重要なことである. 事前に相手の特徴や考え方を調べておくことで, 交渉のさい, スムーズに話合いが進むであろうし, 駆け引きもやりやすくなる. したがって, 過去のデータから交渉者のモデルをつくり学習することは, 重要であると考える.

本研究では交渉の場のかわりに, 繰り返しジャンケンゲームという対戦ゲームを設定し, 対戦結果を利用して対戦相手の戦略モデルを獲得する枠組をつくった. 繰り返しジャンケンゲームは, その名の通りジャンケンを繰り返して対戦するゲームであり, 勝敗は合計得点で決める.[1][2] 1回限りの対戦では, グー, チョキ, パーいずれの手を出しても, 勝負への期待値は変わらない. しかし, 繰り返しジャンケンゲームでは, 同じ相手と繰り返し対戦することになるので, 図1のように, それまでの自分の手と相手の手の情報が蓄積される.

相手が一定の戦略にしたがって次の手を決めているならば, 蓄積された過去の手の情報を利用することで, 相手の次の手を予測することが可能になる. 蓄積された過去の手をもとに次に出す手を決めることを繰り返しジャンケンゲームにおける戦略とする. そして, 過去の手を入力として, 次に

*大学院工学研究科
**工学部知能システム工学科
*Graduate School of engineering
**Dept. of Human and Intelligent systems

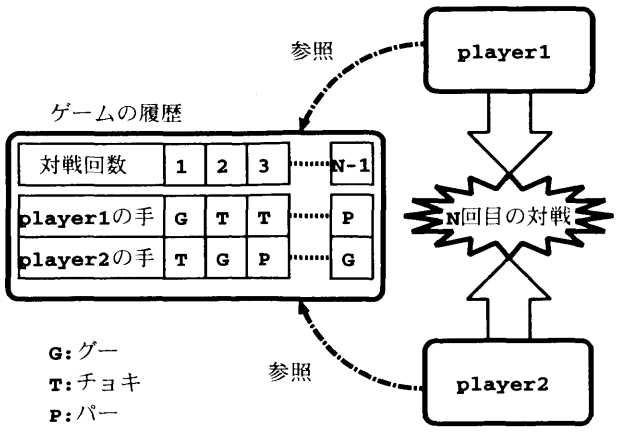

図 1: 対戦の様子

出す手を出力とする関数式で戦略知識を表現する. 対戦相手の戦略モデルを獲得する手法として, 遺伝的プログラミング (GP) を用いた. GP は, 構造的表現を扱えるように遺伝的アルゴリズムを拡張したものである.[3][4] GP を用いることで, 関数式の構造を扱うことができ, 戦略そのものを進化させることができると考えた.

まず, 簡単な戦略をプログラムで作成し, 設定した基本関数群で表現できるか, GP による知識獲得実験をおこなった. その結果, 大幅に勝ち越す個体を獲得することができ, 対戦相手となった戦略に対して有利な戦略知識を獲得できた. また, 獲得した個体の中には, 対戦相手の戦略そのものを内包するものもあった. 次に, 人間の戦略モデルを獲得できるか, 実験をおこなった. 実際には, 人間とコンピュータの戦略が対戦した際の人間が出した手の時系列データをもとに, 知識獲得実験をおこなった. その結果, 対戦相手としたデータだけでなく, 同じ人間の別のデータに対しても有利な個体を獲得することができた. このことから, その人間の戦略モデルといえる知識構造を近似的に獲得できたと思われる.

2 戦略知識の進化的獲得

2.1 戦略知識の関数式表現

RJG の戦略を過去の手の入力として, 次に出す手を出力とする次のような関数式で表現する.

$$n = F(o(t), o(t-1), \cdots m(t), m(t-1))$$

- o(t):t 回前の相手の手
- m(t):t 回前の自分の手

関数式を構成する基本関数を表 1にまとめる. 基本関数はそれぞれ固有の引数を取り, 演算した結果を返す. 基本関数には, 加算や減算といった四則演算をはじめとして,if_guu や,if_paa といった, 条件分岐型の関数を設定した. 表中の my_h,opp_h における初期値は G を設定する.

2.2 GP による知識の獲得

戦略知識の獲得手法として GP を用いる. 評価方法に繰り返しジャンケンゲームの対戦結果を利用することで, 対戦相手に有利な個体に進化させることができる.

2.2.1 遺伝子コーディング

戦略を遺伝子とし, 木構造で表現する. 各個体の木構造のノードは図 2のような構造体で表す.

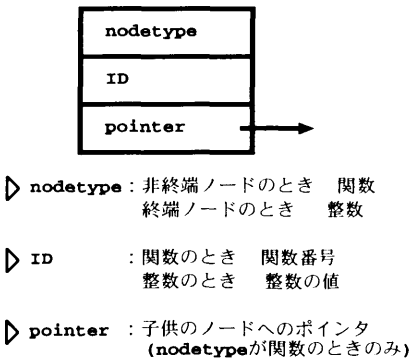

図 2: 木構造のノード

nodetype はノードが関数か整数かを示す要素である. ID は, ノードが持つ値で, ノードが関数の場合はどの関数を使うか示す関数番号を表し, 整数の場合は, その数値を表す. 表 1の ID が各基本関数の関数番号である. pointer はノードが関数のと

表 1: 基本関数の定義

ID	関数名	表示名	機能
0	add(x,y)	+	x+y を返す
1	sub(x,y)	-	x-y を返す
2	multiple(x,y)	*	x*y を返す
3	divide(x,y)	/	x/y を返す x=0 あるいは y=0 の場合は 0 を返す
4	mod(x,y)	%	x を y で割った余りを返す
5	plus1(x)	plus1	x+1 を返す
6	plus2(x)	plus2	x+2 を返す
7	my-hand(x)	my_h	x 手前の自分の手を返す x=0 の場合は 1 手前の自分の手を, x>N（現在の対戦回数）の場合, 初期値を返す
8	opp-hand(x)	opp_h	x 手前の相手の手を返す x=0 の場合は 1 手前の相手の手を, x>N（現在の対戦回数）の場合, 初期値を返す
9	if-guu(x,y1,y2)	if_g	x を 3 で割った余りが 0 の場合 y1 を, 余りが 0 以外の場合 y2 を返す
10	if-tyoki(x,y1,y2)	if_t	x を 3 で割った余りが 1 の場合 y1 を, 余りが 1 以外の場合 y2 を返す
11	if-paa(x,y1,y2)	if_p	x を 3 で割った余りが 2 の場合 y1 を, 余りが 2 以外の場合 y2 を返す

きのみ存在し, 関数がとる引数に応じた数のポインタをもつ. このような構造体を組み合わせ, 木構造で表現したのが図 3である.

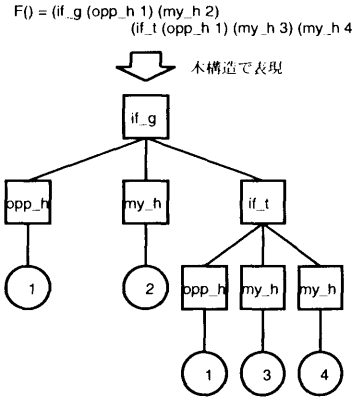

図 3: 戦略知識の木構造表現

この関数式は, 相手の 1 手前の手が G ならば,1 手前に自分が出した手を次に出す手とする. また, 相手の 1 手前の手が T ならば,2 手前に自分が出した手を次の手とする.1 手前の相手の手が,G でも T でもない, つまり,P の場合は,3 手前に自分が出した手を次の手とする戦略知識を表している.

2.2.2 知識獲得システム

図 4に本研究で使用した知識獲得システムの概要を示す.

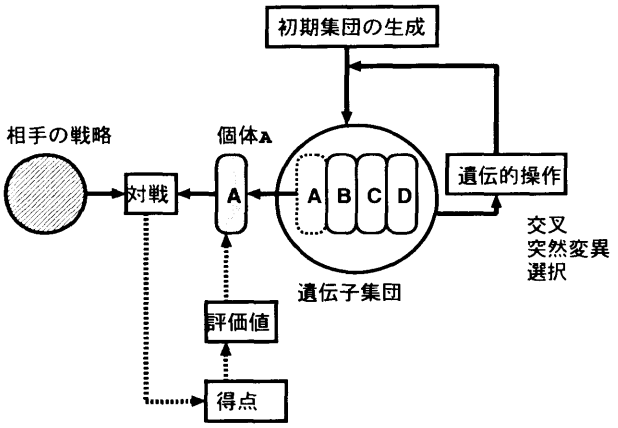

図 4: 対戦の流れ

まず, 初期集団の個体をランダムに生成する. 集団内の各個体は獲得対象となる相手戦略と繰り返しジャンケンゲームをおこない, その得点を評価

値として得る. 全ての個体が試合を行ない, 評価値を得た時点で遺伝的操作をくわえる. そして, 世代を交替し, 次世代の評価に移る.

以上のような流れで打ち切り世代に達するまで個体を進化させ, 繰り返しジャンケンゲームに関する相手の戦略知識の獲得を目指す.

2.2.3 遺伝的操作

次に, 本研究で用いた遺伝的操作について簡単に説明する.

(1) 交叉

選択されたペアの部分木をランダムに選択し, 両者の部分木を取り換え交叉を実行する. 図5に交叉の例を示す. まず, 評価値をもとに親となる個体のペアを選択する. 選択された個体の全ノードの中から, 乱数を使用して交叉ポイントとなるノードを決定する. 図5では, 親1のmy_h, 親2の8が交叉ポイントとなるノードである. 次に, それぞれの交叉ポイントに選ばれたノード以下の部分木を取り換える. 以上の手順で交叉を行なう.

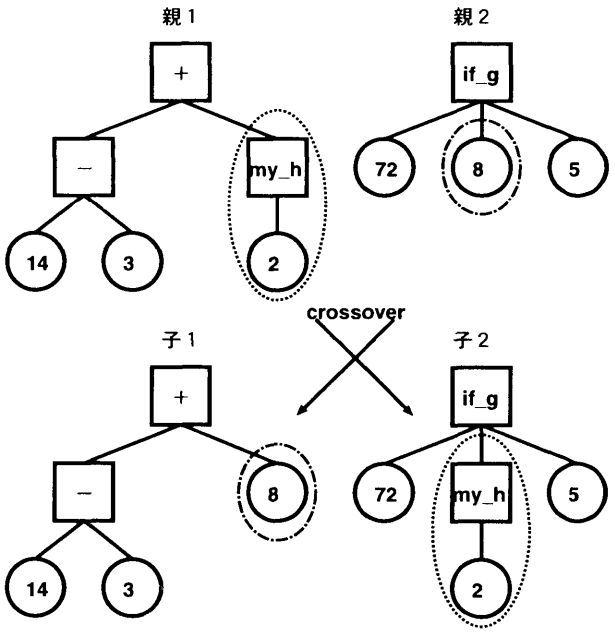

図 5: 交叉の例

(2) 突然変異

交叉後の個体を突然変異率に応じて突然変異させる. 突然変異を行なう場合は, 交叉の時と同じように個体の全ノード数を調べ, 乱数によってあるノードを決める. そして, 新たに生成した部分木をそのノード以下の部分木と置き換える. 新しい部分木の生成は, 初期集団を作る時のアルゴリズムを利用した. 図6に突然変異の例を示す. この場合,opp_h が突然変異をおこなうノードである. そして, 新たに生成した $(+25\ \ (my\_h\ \ 3))$ という部分木を $(opp\_h\ \ 1)$ と置き換え, 突然変異を終了する.

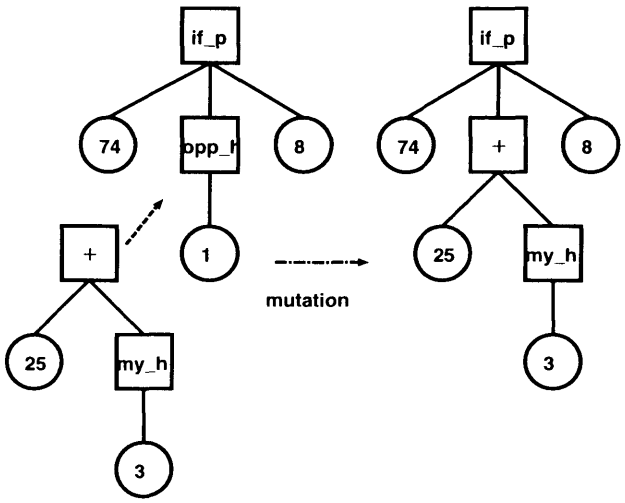

図 6: 突然変異の例

3 知識獲得実験

前章で説明した知識獲得の枠組を用いて実際に知識獲得実験を試みた. まず,簡単な戦略を用意し,本研究で設定した関数群で表現できるか,また,GPによる進化的獲得がうまく機能するか検討した.

獲得の対象となる戦略として仕返し戦略, 繰り返し戦略, 履歴 N 連鎖戦略の 3 種類を用意した. 仕返し戦略は相手が前に出した手をそのまま出す戦略. 繰り返し戦略は"GTP"といったあるパターンを繰り返し出す戦略. そして, 履歴 N 連鎖戦略は相手の手の連鎖の度合をみて次に出す手を決める戦略である. これらの戦略をプログラムでつくり, 繰り返しジャンケンゲームの対戦相手とした.

実験に使用したパラメータを以下に示す.

表 2: 実験のパラメータ

個体数	100
打ち切り世代数	100
突然変異率	0.07
繰り返しジャンケンゲームの対戦回数	1000

評価値は勝ちの総和とする. したがって, 評価値の幅は 0〜1000 となる.

3.1 1 手前仕返し戦略に対する実験結果

1 手前仕返し戦略は相手の 1 回前の手をそのまま出す戦略である. 各世代ごとのエリート評価値をグラフにまとめた. 世代を経るごとに評価値が上昇していることがわかる.

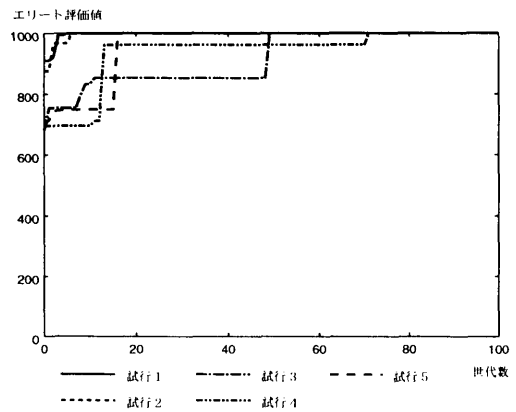

図 7: 各世代ごとのエリート評価値

獲得したエリート個体の 1 つを図 8に示す.

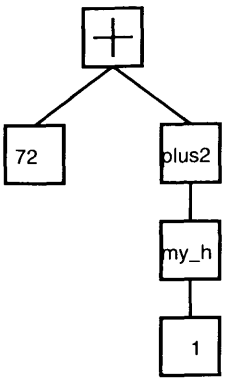

図 8: 獲得したエリート個体

図 8の戦略木は,$(+72(plus2(my\_1)))$ である. 1 手前の自分の手に+2 して,72 を足した値を返す. 次の手を決めるには, その値を mod3 すればいい. ただし,72 は 3 で割り切れるため, 加算しても式の値を 3 で割ったあまりは変わらない. したがって, この最良の個体の戦略を決定付けているのは,$(plus2(my\_1))$ という部分木であることが分かる. (my_1) は相手の戦略そのものであり,plus2 することで優位に立つ戦略になっていることが分かる.

3.2 GPTTP 繰り返し戦略に対する実験結果

繰り返しのパターンは GPTTP とする. 各試行における世代ごとのエリート評価値をグラフにまとめた.

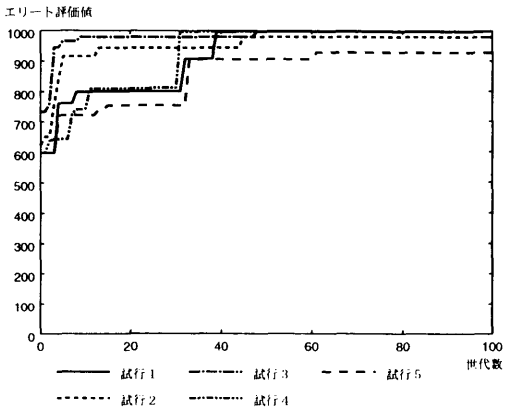

図 9: 各世代ごとのエリート評価値

5 回の試行全てで高い評価値をもつ個体を獲得することができた.

図 10にこの実験で獲得したエリート個体を示す. また, (/ 892 (% 170 224)) を, 演算した結果は5 であるので, 図 11のように簡単化することができる.

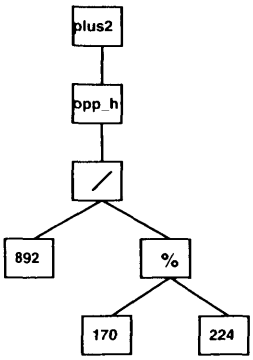

図 10: 獲得したエリート個体

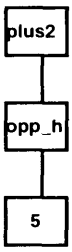

図 11: 簡単化したエリート個体

(*opp_h* 5) は, 相手の 5 手前の手を返す. G,T,P に+2 すると,P,G,T となり, もとの手に勝つ手となるため, 獲得したエリート個体は, 相手の 5 手前の手に勝つ手を出す戦略である. ただし,(*opp_h* 5) は,5 回目の対戦までは初期値である G を返すので, その間は,P をだし続け, 結果最初の GPTTP というパターンに対して 1 勝 2 敗 2 分けとなる. しかし,2 回目以降には PTGGT という手を出すので, 全勝する.

3.3 履歴 2 連鎖戦略に対する実験結果

履歴 N 連鎖戦略は, ゲームの履歴をより有効に活用する戦略である. ゲームの履歴として蓄積していた相手の手の連鎖の度合をみて, 次に出す手を決める戦略である. 表 3で説明すると, 例えば, 相手が G,T と出した後に,P を出した場合, GTP の項目を+1 する. このようにして, 相手の手の連鎖の度合をパターン毎に調べ, 履歴連鎖テーブルを構築する. このテーブルを使う事で様々な戦略が考えられるが, 本研究では, 基本的に, 一番多い連鎖につながる手を出して来ると予想する戦略を履歴 N 連鎖戦略とする. 相手が T,P と出して来た場合, 表 3をみると, TPG の項目が最も多いので, 相手は G を出して来ると予想し, こちらは P を出す.

連鎖数 N を増やせば, より細かいパターンを分類できるが, 連鎖の度合が分散し, また, パターンを検出するのにより多くの対戦回数を必要とするため,N を増やすほど強い戦略ができるとは一概にいえない. ここでは, 履歴 2 連鎖戦略に対して, 知識獲得実験をおこなった.

表 3: 履歴連鎖テーブル

O_{n-2}	O_{n-1}	O_n	回数
G	G	G	0
⋮	⋮	⋮	⋮
G	T	P	1
⋮	⋮	⋮	⋮
T	P	G	3
T	P	T	0
T	P	P	2
⋮	⋮	⋮	⋮
P	P	P	0

各世代ごとのエリート評価値をグラフにまとめた.

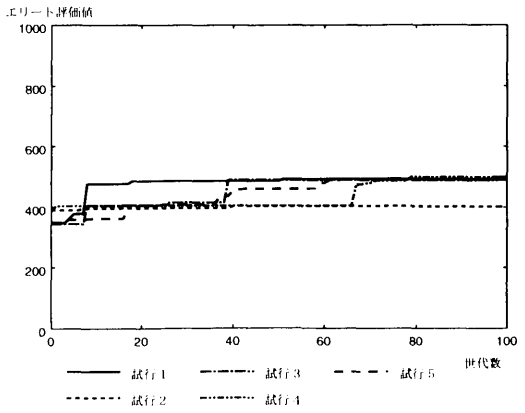

図 12: 各世代のエリート評価値

表 4: エリート個体の勝敗数

エリート個体	勝ち	敗け	引き分け
試行 1	491	485	24
試行 2	402	401	197
試行 3	498	414	88
試行 4	493	497	10
試行 5	489	487	24

実験の結果, 互角の勝負をする戦略知識を獲得したものの, 大幅に勝ち越すことはできず, 履歴 N 連鎖戦略に対して有効な戦略知識を獲得することはできなかった.

獲得したエリート個体の一つを図 13に示す.

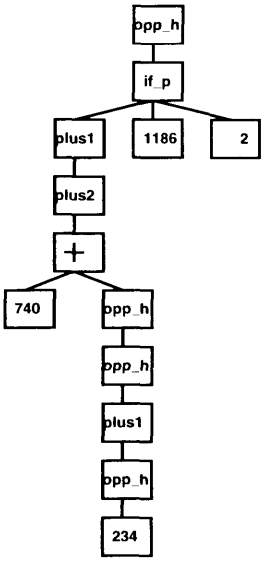

図 13: 獲得したエリート個体

この個体は,if_paa の条件部である (plus1 (plus2 …以下の部分木を 3 で割ったあまりが, P の値 2 を返すならば,1186 手前の相手の手を,2 以外の値ならば 2 手前の相手の手を出す戦略知識である. 対戦回数は 1000 回なので,1186 手前の相手の手は存在しない. この場合, 初期値である G を出す.

3.4 人間の戦略モデルの獲得

本研究では, これまで, プログラムで生成した戦略を相手として繰り返しジャンケンゲームの試合をおこない, そのモデルを獲得する実験をおこなってきた. その結果, 履歴戦略には余り通用しなかったが, 他の固定的な戦略を近似的に表現することができた. そこで, 今度は人間を対象とし, その戦略モデルを獲得できるか試みた.

3.4.1 実験方法

実験は直接人間と対戦し, その過程で相手の戦略モデルを獲得するのではなく, すでにある対戦結果をもとに学習をかけて, 人間の戦略モデル獲得を試みた. そのため, 知識獲得実験を行うにあたって, 人間にコンピュータと対戦してもらった. コンピュータ側の戦略は履歴 2 連鎖戦略を使用した. 30 回の対戦を 6 試合分おこない, 1 試合目の人間の手を data1, 2 試合目を data2 として data6 まで保存した.

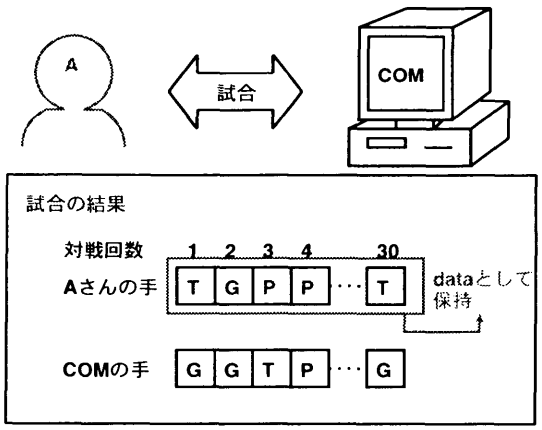

図 14: 人間 vs コンピュータ

各 data は人間の手の時系列データであり, この順番に手を出す戦略とする. これらのデータを用いることで, 疑似的に人間と対戦し, 人間が出した手のモデルが本研究で設定した関数式で表現できるか検討する.

具体的な実験方法を図 15に示す.

まず,6 つのデータから 1 つ data1 を獲得対象戦略として設定する. data1 と集団内の各個体で繰り返しジャンケンゲームをおこない, 得た得点を評価値として個体に与える. その評価値をもとに遺伝的操作を加え, 世代を交替する. これらの処理を打ち切り世代に達するまで繰り返し, data1 に対して有効な戦略知識の獲得を試みる.

次に, 知識獲得実験で得られたエリート個体をその他の 5 つのデータと対戦させ, エリート個体の能力を検証する.

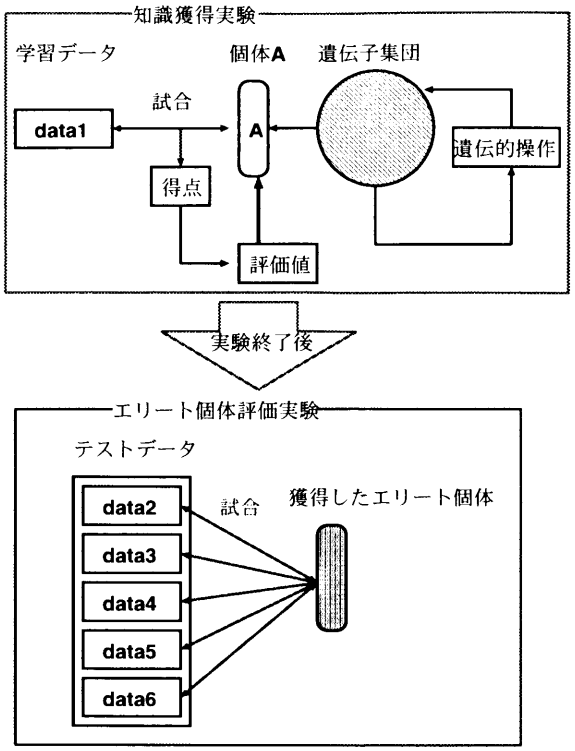

図 15: 実験方法

実験に使用したパラメータを以下に示す.

表 5: 実験のパラメータ

個体数	100
打ち切り世代数	100
突然変異率	0.07
繰り返しジャンケンゲームの対戦回数	30

評価値は勝ち数の総和を与えるので, 評価値の幅は 0〜30 となる.

3.4.2 実験結果

知識獲得実験の結果として, 世代ごとのエリート評価値と平均評価値を図 16のグラフに示す. 世代が進むごとに, エリート個体の評価値も上昇し, 平均値も上昇していることから, 集団内に相手の戦略に対して有効な知識構造がひろまったといえる.

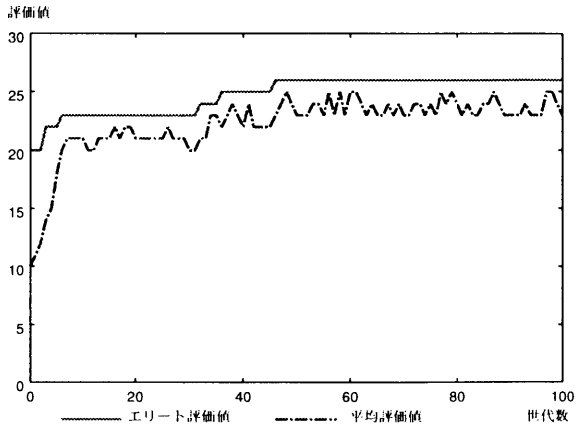

図 16: data1 に対する実験結果

実験で獲得したエリート個体の性能を評価するため, 獲得の対象としたデータ以外の各データと繰り返しジャンケンゲームの試合を行なった. その結果を表 6にまとめた.

この実験では, 人間の手のデータ列を対戦相手とし, その知識獲得を試みた. 実験の結果, 対戦相手としたデータ列に対して高い勝率を得た個体を獲得することができた. したがって, 人間が出した手を本研究で設定した関数式で表現することができたといえる. また, 表 6をみると, 獲得したエリートは, 獲得の対象となった以外のデータに対しても勝ち越す場合がある. これは, 各データに共通する何らかの特徴を知識として獲得できたものと考えられる.

data1 を獲得対象とした実験で獲得したエリート個体の一つを, 図 17に示す. 先程のプログラムの戦略を獲得対象にした実験で得たエリート個体よりも, はるかに複雑で大きな構造をもつ個体を獲得した.

表 6: 獲得したエリート個体の対戦結果

対戦相手	data1			data2			data3			data4			data5			data6		
学習データ	勝	敗	分	勝	敗	分	勝	敗	分	勝	敗	分	勝	敗	分	勝	敗	分
data1				15	12	3	18	7	5	16	11	3	11	5	14	12	6	12
data2	11	6	13				17	5	8	18	2	10	2	16	12	8	10	12
data3	14	11	5	15	11	4				12	13	5	10	4	16	10	7	13
data4	7	6	17	17	4	9	10	5	15				3	18	14	7	11	12
data5	8	11	11	9	11	10	14	7	9	16	6	8				12	3	15
data6	15	13	2	14	10	6	12	11	7	8	16	6	8	14	8			

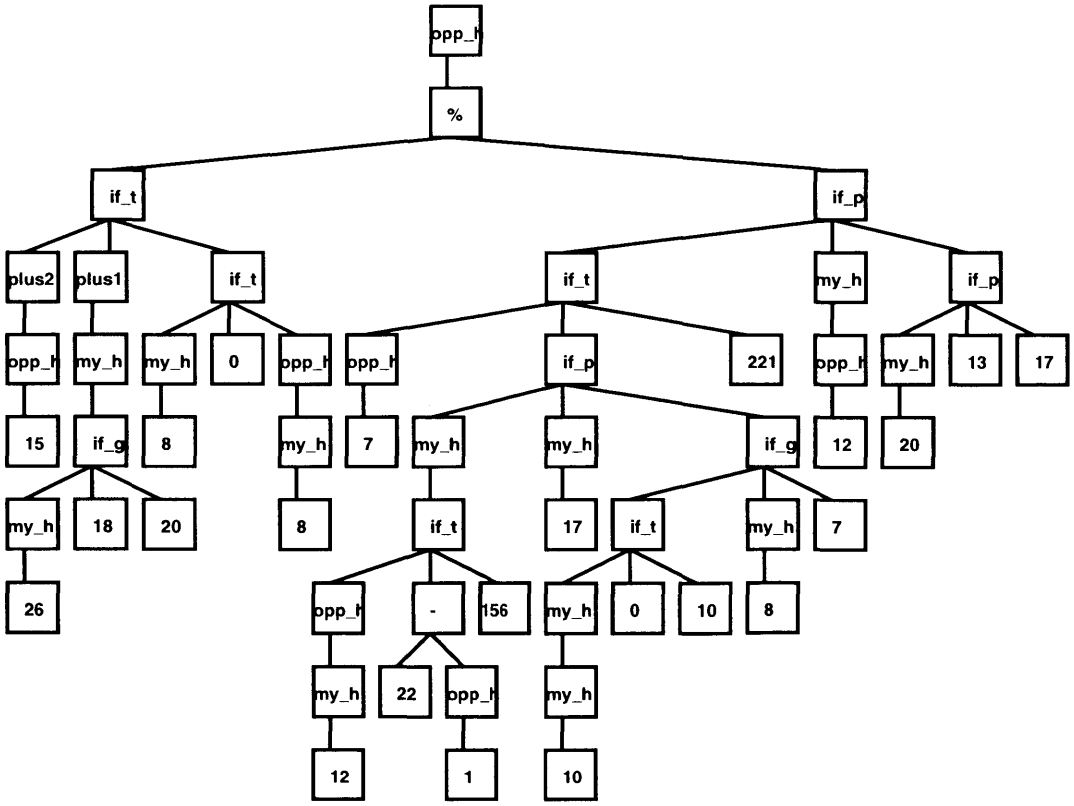

図 17: data1 のエリート個体

4 考察

本研究で設定した基本関数群と整数による関数式でどういった戦略知識が表現できるのか, まず, プログラムで生成した簡単な戦略を相手にして知識獲得をおこなった.

1 手前仕返し戦略に対する実験では,$(my\_h\ 1)$ という部分木を持つ個体を獲得した. この部分木は自分の 1 手前の手を返す. すなわち, 1 手前仕返し戦略そのものである. このように, 相手の戦略そのものを内包するような個体の獲得に成功している.

周期 5 の GPTTP 繰り返し戦略に対する実験結果では, $(plus2\ (opp\_h\ 5))$ というエリート個体を獲得した. この個体は,5 手前の相手の手に勝つ手を出す戦略で, "周期 5 の繰り返し"という考え方を知識として獲得できており, 獲得対象とした GPTTP 繰り返し戦略のみならず, 周期 5 の繰り返し戦略全てに対して有効な戦略知識といえる. 履歴 2 連鎖戦略を獲得対象とした実験では, エリート評価値がほぼ半分にまでしか上昇せず, 有効な戦略知識の獲得には至らなかった. ただし, 全ての対戦結果を参照する履歴 2 連鎖戦略に対して, 互角の勝負をする個体を深さ 10 程度の木で表現できたことは評価できると思われる.

以上の結果から, 履歴 2 連鎖戦略はうまく表現できなかったが, その他の簡単な戦略に対しては, 遺伝的プログラミングによって関数群がうまく組み合わさり, 獲得対象の戦略を表現できることがわかった.

次に, 人間がもつ戦略知識を獲得できるか実験を行なった. 人間に 1000 回ものジャンケンをしてもらうのは, 不可能ではないが, 後半面倒になって同じ手を出し続けるなどの問題点があり, 実現は困難である. したがって, 今回は履歴 2 連鎖戦略と 30 回の対戦をしてもらい, その対戦結果から, 人間側の手の時系列データを使って知識獲得実験を行なった.

その結果, 人間が出した手の系列に対して有効な手を出す個体を獲得することができた. また, 獲得したエリート個体は, 獲得対象以外の未知のデータに対しても有効な場合があった. このことから, 複数のデータに共通して出ていた特徴, すなわちその人のくせや, 考え方といったものを知識として獲得できたと思われる. 今回の実験によって, 人間が出した手について本論文で設定した関数群で表現することが可能だとわかった.

5 まとめと今後の課題

本論文では, 繰り返しジャンケンゲームにおける戦略を獲得する手法について検討してきた. まず, 戦略を知識として一般化するために, 過去の手を入力として次の手を決定する関数式を用いて戦略を表現した. この関数式は基本的な関数群と整数の組合せで構成され,S 式で表した. 次に, 獲得する手法として GP を用いた.GP によって戦略の構造自体を変化させ, 対戦相手の戦略に適した知識構造を持つ個体に進化させることができると考えた. 実際に, プログラムで生成した簡単な戦略を対戦相手とした場合, 相手の戦略の根幹となる部分を知識構造として獲得することができた.

また, 人間の手の時系列データを対戦相手として知識獲得実験を行なった結果, 人間が出した手に対して有効な手を出す戦略知識を獲得できた. また, その個体は, 対戦相手以外の未知のデータに対してもある程度いい対戦結果を得ることができた.

今後は, 実際に人間と対戦しながら戦略を進化させ, 対戦相手の戦略モデルを獲得する実験をおこなう. そのためには, 少ないデータからいかに相手のモデルを獲得するかを検討しなければならない. また,GP による進化には時間がかかるため, 少ない時間で効率良く学習する手法を検討する必要がある.

参考文献

[1] 弓削孝文, 西野順二, 小高知宏, 小倉久和, 繰り返しジャンケンゲーム戦略知識の進化的獲得, 平成 10 年度卒業論文.

[2] 牧野泰裕, 西野順二, 小高知宏, 小倉久和, 遺伝的アルゴリズムによる対戦型ゲーム戦略の獲得, 平成 8 年度修士論文.

[3] 伊庭 斉志, 遺伝的プログラミング, 東京電気大学出版局,1996.

[4] 伊庭 斉志, 遺伝的アルゴリズムの基礎, オーム社,1994.

[5] 北野 宏明, 遺伝的アルゴリズム, 産業図書,1993.

[6] 鈴木光男, 新ゲーム理論, 勁草書房,1990.

[7] R.S.Michalski 他編, 電総研人工知能研究グループ訳, 知識獲得入門 帰納学習と応用, 共立出版株式会社,1987.

[8] 淵一博監修, 古川康一・溝口文雄共編, 知識の学習メカニズム, 共立出版株式会社,1986.

[9] Philip D.Laird 著, 横森貴訳, 例からの学習ー計算論的学習理論ー, オーム社,1992.

[10] 弓削孝文, 西野順二, 小高知宏, 小倉久和, 繰り返しジャンケンゲームにおける戦略知識の表現と進化的獲得, 日本機械学会第 9 回インテリジェント・システム・シンポジウム講演論文集, pp.271–pp.274,2000.

[11] 弓削孝文, 西野順二, 小高知宏, 小倉久和, 繰り返しジャンケンゲームにおける戦略知識の表現と進化的獲得, 第 60 回情報処理学会全国大会講演論文集, pp.2/325–326,2000.

[12] 弓削孝文, 西野順二, 小高知宏, 小倉久和, 繰り返しジャンケンゲームにおける戦略知識の関数式表現と進化的獲得, 電気関連学会北陸支部連合大会 (地方会) P.418,2000.